

IN THE CLAIMS:

1. (Currently Amended): A method for creating a software state machine, comprising:

providing a state machine object in a runtime environment on a computer hardware platform, wherein the state machine object includes an object constructor method; and

providing an initializer object in the runtime environment on the computer hardware platform, wherein the initializer object defines states, actions, and conditions for a software state machine and wherein the object constructor method is configured to create an instance of the initializer object.

wherein the state machine object is configured to use the initializer object to create a table object, wherein the initializer object includes a table element array creation method and wherein the state machine object is configured to call the table element array creation method to create the table object using the results of the table element array creation method.

wherein the state machine object is configured to create an array of state variables, wherein the initializer object includes a table variable array creation method and wherein the state machine object is configured to call the table variable array creation method to create the array of state variables using the results of the table variable array creation method.

wherein the table object is configured to create an array of state transition objects based on the array of state variables and return the array of state transition objects to the state machine object, and

wherein the state machine object is configured to execute the software state machine using the array of state transition objects.

2-8. (Canceled)

9. (Original): The method of claim 1, wherein at least one of the state machine object and the initializer object implements an interface.

10. (Original): The method of claim 1, wherein the state machine object includes a state method that is configured to return a current state of the software state machine.

11. (Currently Amended): A method for creating software state machines, comprising:

providing a state machine object in a runtime environment on a computer hardware platform;

receiving, using a set of graphical user interfaces in the runtime environment on the computer hardware platform, entry of a plurality of states, one or more actions, one or more inputs, one or more conditions, one or more events, one or more triggers, and a plurality of state transitions for a software state machine;

providing creating an initializer object in the runtime environment on the computer hardware platform, wherein the initializer object defines [[a]] the plurality of states, the one or more actions, the one or more inputs, the one or more conditions, the one or more events, [[and]] the one or more triggers, and the plurality of state transitions for [[a]] the software state machine, wherein:

configuring the state machine object in the runtime environment on the computer hardware platform is configured to use the initializer object to create an array of state transition objects based on the plurality of state transitions, wherein each state transition object in the array of state transition objects defines references at least one of the one or more conditions that causes a given state transition in the software state machine, and wherein each condition is a Boolean expression formed from at least one of the one or more inputs; and

responsive to occurrence of a trigger, evaluating the one or more inputs, computing the one or more conditions, and determining a next state based on the array of state transition objects,

wherein a first state transition object in the array of state transition objects defines an action to take responsive to a given state transition, and

wherein a second state transition object in the array of state transition objects defines an event to be generated responsive to a given state transition.

12. (Currently Amended): An apparatus for creating a software state machine, comprising:

a processor; and

a memory having stored therein a state machine object and an initializer object, wherein the state machine object includes an object constructor method,

wherein the initializer object defines states, actions, and conditions for a state machine,
wherein the object constructor method is configured to create an instance of the initializer object,

wherein the state machine object is configured to use the initializer object to create a table object, wherein the initializer object includes a table element array creation method and wherein the state machine object is configured to call the table element array creation method to create the table object using the results of the table element array creation method.

wherein the state machine object is configured to create an array of state variables, wherein the initializer object includes a table variable array creation method and wherein the state machine object is configured to call the table variable array creation method to create the array of state variables using the results of the variable array creation method.

wherein the table object is configured to create an array of state transition objects based on the array of state variables and return the array of state transition objects to the state machine object, and

wherein the state machine object is configured to execute the software state machine using the array of state transition objects.

13-19. (Canceled)

20. (Original): The apparatus of claim 12, wherein at least one of the state machine object and the initializer object implements an interface.

21. (Original): The apparatus of claim 12, wherein the state machine object includes a state method that is configured to return a current state of the software state machine.

22. (Canceled)

23. (Currently Amended): A computer program product, in a tangible computer readable medium, for creating a software state machine, comprising:

instructions for providing a state machine object, wherein the state machine object includes an object constructor method; and

instructions for providing an initializer object, wherein the initializer object defines states, actions, and conditions for a software state machine, wherein the object constructor method is configured to create an instance of the initializer object,

wherein the state machine object is configured to use the initializer object to create a table object, wherein the initializer object includes a table element array creation method and wherein the state machine object is configured to call the table element array creation method to create the table object using the results of the table element array creation method,

wherein the table object is configured to create an array of state variables, wherein the initializer object includes a table variable array creation method and wherein the state machine object is configured to call the table variable array creation method to create the array of state variables using the results of the variable array creation method,

wherein the table object is configured to create an array of state transition objects based on the array of state variables and return the array of state transition objects to the state machine object,

wherein the state machine object is configured to receive the array of state transition objects and execute the software state machine using the array of state transition objects.

24. (Currently Amended): A computer program product, in a tangible computer readable medium, for creating software state machines, comprising:

instructions for providing a state machine object;

instructions for receiving, using a set of graphical user interfaces, entry of a plurality of states, one or more actions, one or more inputs, one or more conditions, one or more events, one or more triggers, and a plurality of state transitions for a software state machine;

instructions for ~~providing~~ creating an initializer object, wherein the initializer object defines [[a]] the plurality of states, the one or more actions, the one or more inputs, the one or more conditions, the one or more events, [[and]] the one or more triggers, and the plurality of state transitions for [[a]] the software state machine, wherein;

configuring the state machine object ~~is configured~~ to use the initializer object to create an array of state transition objects based on the plurality of state transitions, wherein each state transition object in the array of state transition objects defines references at least one of the one or more conditions that causes a given state transition in the software state machine, and wherein

each condition is a Boolean expression formed from at least one of the one or more inputs; and instructions, responsive to occurrence of a trigger, for evaluating the one or more inputs, computing the one or more conditions, and determining a next state based on the array of state transition objects,

wherein a first state transition object in the array of state transition objects defines an action to take responsive to a given state transition, and

wherein a second state transition object in the array of state transition objects defines an event to be generated responsive to a given state transition.

25-30. (Canceled)